UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
10/608,039	06/30/2003	Sanjay Ghemawat	0026-0030	7559

44989 7590 05/11/2010

HARRITY & HARRITY, LLP
11350 Random Hills Road
SUITE 600
FAIRFAX, VA 22030

EXAMINER	
DAYE, CHELCIE L	

ART UNIT	PAPER NUMBER
2161	

MAIL DATE	DELIVERY MODE
05/11/2010	PAPER

Please find below and/or attached an Office communication concerning this application or proceeding.

The time period for reply, if any, is set in the attached communication.

UNITED STATES PATENT AND TRADEMARK OFFICE

BEFORE THE BOARD OF PATENT APPEALS
AND INTERFERENCES

Ex parte SANJAY GHEMAWAT, HOWARD GOBIOFF,
and SHUN-TAK LEUNG

Appeal 2008-002267
Application 10/608,039
Technology Center 2100

Decided: May 11, 2010

Before MAHSHID D. SAADAT, MARC S. HOFF, and
BRADLEY W. BAUMEISTER, *Administrative Patent Judges*.

BAUMEISTER, *Administrative Patent Judge*.

DECISION ON APPEAL

STATEMENT OF CASE

Appellants appeal under 35 U.S.C. § 134 from the Examiner's rejection of claims 1-25. We have jurisdiction under 35 U.S.C. § 6(b).

We REVERSE.

Appellants' invention relates to "[a computer] system [that] facilitates the deletion of data, such as files, orphaned chunks, and stale replicas" (Abstract).

Claims 1, 2, 4-7, 10-13, and 20-25 stand rejected under 35 U.S.C. § 102(b) as anticipated by Mattis (US 6,209,003 B1).

Claim 3 stands rejected under 35 U.S.C. § 103(a) as obvious over Mattis in view of Manley (US 2003/0182330 A1).

Claims 8, 9, and 14-19 stand rejected under 35 U.S.C. § 103(a) as obvious over Mattis in view of ANDY HISGEN ET AL., NEW-VALUE LOGGING IN THE ECHO REPLICATED FILE SYSTEM (Digital Equip. Corp. 1993).

FACTUAL FINDINGS

Mattis

1. Mattis discloses a garbage collection process (col. 21, l. 48 – col. 26, l. 22) "to periodically clean up cache storage areas without interrupting or affecting the operation of the cache" (col. 21, ll. 56-57). The garbage collecting process is an "evacuation type" in which target areas or "arenas" of the cache are scanned, various data fragments of an arena are identified as being active and moved to other arenas, and other data fragments are determined to be subject to deletion, thereby leaving the arena empty at the end of the process (col. 21, l. 59 – col. 22, l. 7). Fragments are deleted by

marking them as deleted and overwriting the data in the fragment (col. 23, ll. 15-17).

2. Mattis defines the term "object" as "a network resource or any discrete element of information that is delivered from a server" (col. 7, ll. 23-25).

3. Mattis discloses using a read counter to indicate whether an information object has been loaded recently (col. 26, ll. 8-22).

4.

> Each pool header 202 stores a Magic number, a Version No. value, a No. of Arenas value, and one or more arena headers 206*a*-206*n*. The Magic number is used solely for internal consistency checks. The Version No. value stores a version number of the program or process that created the arenas 206*a*-206*n* in the pool. It is used for consistency checks to ensure that the currently executing version of the cache 80 can properly read and write the arenas.

(Col. 17, ll. 39-46).

I.

Claims 1-13

Independent claim 1 is illustrative of the aspect of Appellants' invention in which a master first receives, from chunk servers, information concerning chunks stored by the chunk servers, and subsequently identifies, to one of these servers, one of the chunks that corresponds to a file that has been previously permanently deleted after being renamed (*see, e.g.*, Spec. [0071]-[0072]):

> 1. A method for deleting one or more of a plurality of files, the files including one or more chunks stored by a plurality of servers, the method comprising:
>
> identifying a file to be deleted;

> renaming the identified file;
>
> permanently deleting the renamed file a predetermined amount of time after renaming the identified file as part of a garbage collection process;
>
> receiving, from the servers, information concerning chunks stored by the servers; and
>
> identifying, to one of the servers, one of the chunks that corresponds to the permanently deleted file.

The Examiner finds that the combined portions of Mattis – column 23, lines 15-37; column 7, lines 23-29; and Figure 5 – disclose the claim language: "identifying, to one of the servers, one of the chunks that corresponds to the permanently deleted file" (Ans. 12-13).[1] Appellants argue that neither the cited passages nor any other section of Mattis discloses or suggests this claim language (Reply Br. 9-11).

ISSUE

The first issue before us, then, is: Does Mattis disclose a component that, in response to receiving information from the servers concerning chunks stored by the servers, identifies, to one of the servers, one of the chunks that corresponds to one of the one or more permanently deleted chunks?

[1] Rather than repeat the Examiner's positions and Appellants' arguments in their entirety, we refer to the following documents throughout this opinion for their respective details: (1) the Appeal Brief ("App. Br.") filed December 13, 2006; (2) the Examiner's Answer ("Ans.") mailed July 23, 2007; and (3) the Reply Brief ("Reply Br.") filed September 24, 2007.

ANALYSIS

We agree with Appellants that the portions of Mattis cited by the Examiner do not disclose identifying, to one of the servers, one of the chunks that corresponds to a file that has been permanently deleted after being renamed (*id.*). The cited portion of Mattis appearing within column 23 merely describes a garbage collection process in which a fragment in the cache is marked as deleted and its data is subsequently overwritten (Fact 1). The cited portion of column 7 merely defines an object as "a network resource or any discrete element of information that is delivered from a server" (Fact 2). The portions of Mattis relied upon by the Examiner do not disclose any component or process that, in response to receiving information from the servers concerning chunks stored by the servers, identifies, to one of the servers, one of the chunks that corresponds to one of the one or more permanently deleted chunks. The Examiner has not pointed to any other portion of Mattis, nor do we find any teachings, indicating this limitation.

Accordingly, we will reverse the Examiner's rejection of independent claim 1; claims 2-11, which depend from claim 1; and independent claims 12 and 13, both of which contain similar limitations.

II.

Claims 20-25

Independent claim 20 is illustrative of the aspect of Appellants' invention in which the master associates version information with replicas of the data chunks, identifies stale replicas based on the associated version information, and deletes stale replicas (*see, e.g.*, Spec. [0075]-[0077]):

> 20. A method for deleting stale replicas of chunks, the replicas being stored by a plurality of servers, the method comprising:
>
> associating version information with replicas of chunks;
>
> identifying stale replicas based on the associated version information;
>
> deleting the stale replicas;
>
> receiving, from the servers, information concerning replicas stored by the servers; and
>
> identifying, to one of the servers, one of the replicas that corresponds to one of the deleted stale replicas.

The Examiner cites column 17, lines 39-46, and column 26, lines 8-22, of Mattis as disclosing the claimed process for deleting stale replicas (Ans. 17-18). Appellants argue, *inter alia*, that Mattis fails to "disclose or suggest maintaining versions of chunks, let alone identifying a stale chunk based on the versions of the chunks . . . . Instead, Mattis . . . simply uses a high read counter value to reflect an information object that has been recently loaded" (Reply Br. 15-16 (emphases omitted)).

ISSUE

The second issue before us, then, is: Does Mattis disclose maintaining versions of chunks and identifying a stale chunk based on the versions of the chunks?

ANALYSIS

Appellants' arguments are persuasive. Mattis discloses, at column 26, using a read counter in a re-validation process to indicate whether an information object has been loaded recently (Fact 3). The disclosure relating to the re-validation process does not mention maintaining versions of chunks. Rather, Mattis only uses version numbers for the program or process that created the arenas 206*a*-206*n* in the cache's pool (Fact 4).

Accordingly, we will reverse the Examiner's rejection of independent claim 20; claims 21-23, which depend from claim 20; and independent claims 24 and 25, both of which contain similar limitations.

III.

Claim 14-19

Independent claim 14 is illustrative of the aspect of Appellants' invention in which the master maps file names to chunks, identifies and deletes orphaned chunks, receives information from the servers concerning the servers' chunks, and informs the servers that various chunks are orphans (*see, e.g.*, Spec. [0073]-[0074]):

> 14. A method for deleting orphaned chunks of a plurality of chunks stored by a plurality of servers, the method comprising:
>
> providing a mapping of file names to chunks;
>
> identifying chunks, as orphaned chunks, that are not reachable from any of the file names;
>
> deleting the orphaned chunks;
>
> receiving, from the servers, information concerning chunks stored by the servers; and
>
> identifying, to one of the servers, one of the chunks that corresponds to one of the deleted orphaned chunks.

The Examiner finds that Mattis fails to disclose the following steps of claim 14: (1) mapping file names to chunks; (2) identifying certain chunks as orphaned chunks; and (3) deleting the orphaned chunks (Ans. 8-9). The Examiner further finds, though, that Hisgen discloses these steps at pages

23-24, and that motivation existed to combine these teachings with Mattis (Ans. 9).

Appellants argue that even assuming Hisgen discloses a list of orphan files, "the Examiner still has not established that Hisgen . . . discloses or suggest identifying, to one of the servers from which information concerning chunks stored by the servers was received, one of the chunks that corresponds to one of the deleted orphaned chunks, as required by claim 14" (Reply Br. 21-22 (emphasis omitted)).

ISSUE

The third issue before us, then, is: Does the cited prior art disclose or suggest identifying, to one of the servers from which information concerning chunks stored by the servers was received, one of the chunks that corresponds to one of the deleted orphaned chunks?

ANALYSIS

Appellants' arguments are persuasive. The passages of Hisgen relied upon by the Examiner respectively set forth (1) a file identification (fid) map hash table (Hisgen, 23:11-19) and an orphan list (Hisgen, 24:12-26). However, neither of these cited portions teaches or suggests a component that performs the claimed steps of "receiving, from the servers, information concerning chunks stored by the servers; and identifying, to one of the servers, one of the chunks that corresponds to one of the deleted orphaned chunks." Nor do we find these steps either elsewhere in Hisgen or in Mattis.

Accordingly, we will reverse the Examiner's rejection of independent claim 14; claims 15-17, which depend from claim 14; and independent claims 18 and 19, both of which contain similar limitations.

DECISION

We reverse the Examiner's decision rejecting claims 1-25 under 35 U.S.C. §§ 102 and 103.

ORDER

REVERSED

babc

HARRITY & HARRITY, LLP
11350 Random Hills Road
SUITE 600
FAIRFAX, VA 22030